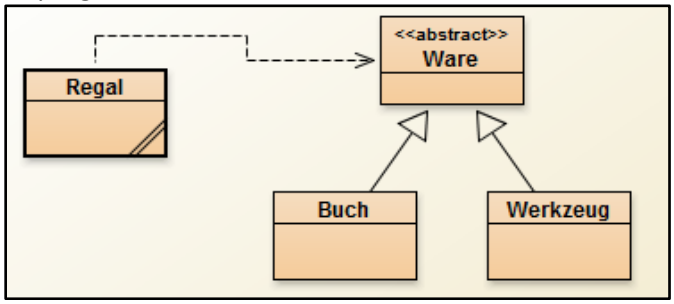
# Shelving system

To get started in C #, a simple console program for a shelving system is to be designed.
The UML-diagrams are taken from a previous Java-exercise. Adjust the names for variables/methods/classes to English ones.
On the shelf two different goods can be stored, namely books and tools.

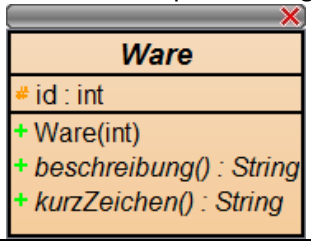
```
Fach 2/3 besetzt --> #3: Werkzeug 9.1 kg
Fach 5/4 besetzt --> #8: Buch 20x16x6

---------------------
|   |   |   |   |   |
---------------------
|   |   |   |   | B |
---------------------
|   |   | W | W |   |
---------------------
|   | B |   |   |   |
---------------------
|   | W |   |   | B |
---------------------
|   |   |   |   | B |
---------------------
Auslastung: 23.33 %
```

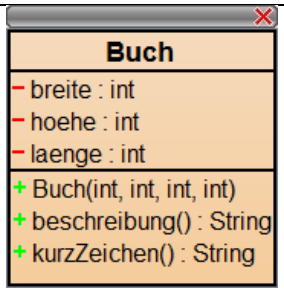
The following classes are to be programmed:




Please note:

- Use suitable constructors
- Where possible, use Formatted Strings
- The instance variables are to be programmed as properties
- Pay attention to the principle of information hiding

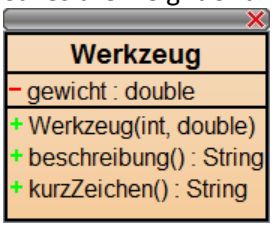
| | |
|---|---|
| **1** | **Ware**<br>Saves the unique ID of a good.<br>  |
| **2** | **Buch**<br>Saves length, width, height of a book. |




**kurzZeichen**: returns „B“
**beschreibung**: returns the Id and the dimensions in the form „#8: Buch 20x16x6„

## 3 Werkzeug

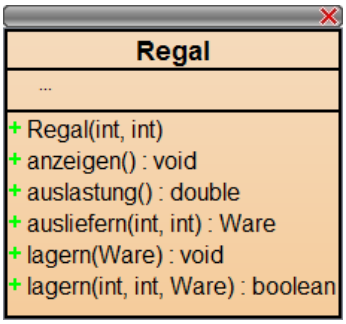
Saves the weight of the tool.




**kurzZeichen**: returns „W“
**beschreibung**: returns the Id and the dimensions in the form „#3: Werkzeug 9.1 kg„.

## 4 Regal

Stores goods in a two-dimensional array. The size of the shelf (number of rows and columns) is specified in the constructor.




Program the following methods:

- **anzeigen**: outputs the shelf as shown in the example above. The short description of the goods is issued for each occupied compartment. The utilization has also to printed to the console.
- **lagern**: stores a product at the specified position. If the tray is already occupied, an error message will be displayed as shown in the screenshot. The return value indicates whether the storage was successful or not.
- **lagern**: the method without position stores the goods at the next free position.
- **ausliefern**: removes the good at the given position.
- **auslastung**: returns the utilization of the shelf in percent.

## 5 Klasse Regalsystem

- **Main**
  Create a shelf of a certain size and store at least 3 tools and at least 4 books. Finally, the shelf is to be displayed.
- **Testmethod**
  Write a test method (just create a usual method called by Main) that creates a new 6x5 shelf and stores the goods from the file **waren.csv** on that shelf. (This file is included in the exercise as zip - add it to the project and set "**Copy to Output Directory**" to "**Copy if newer**"). To read the user's input, use Console.ReadLine () and String-Split(). To read the file use File.ReadAllLines(). The load must then be 23.33%.
  The columns mean: **type; row; col; id; [weight | length; width; height]**